1996年度日本オペレーションズ・リサーチ学会
秋季研究発表会

A Branch and Bound Approach to Two-Stage Hybrid Flowshop Scheduling with Job Splitting and Overlapping Production

02002990 早稲田大学 *今泉 淳 IMAIZUMI Jun
01603200 早稲田大学 森戸 晋 MORITO Susumu

1 はじめに

本研究では、多くのスケジューリング·モデルが想定する機械産業とは異なる特徴を有する、装置産業のシステムを念頭に置いたスケジューリング問題を組合せ最適化問題として扱い、分枝限定法に基づく解法を構築し、実験的に評価する。

具体的には、**1)**2工程のフローショップで、第1、第2工程はそれぞれU台、D台の並列機械からなる、**2)** ジョブは連続的に流れており、第1工程で作業を終了したジョブは逐次**中間品**として溜り、その一部を使って第2工程の作業を開始することができる、すなわち、同一ジョブの第1工程の作業と第2工程の作業が時間軸上で重なって生産されること (重複生産、overlapping production) を許す、**3)** 各ジョブは第2工程で複数の作業に分岐する、などの特徴がある。各作業は、加工される機械と加工時間 (日単位)、中間品に換算した量が所与であり、作業の中断は許されない。

このとき、総所要時間を最小にするように、各作業の開始時刻 (開始日) を決定する (スケジュールの例は図1を参照)。なお、時間軸は第1日から開始し、日単位でスケジュールを考える。

2 記号の定義

あるジョブの第1工程の作業を「**親作業**」、第2工程の作業を「**子作業**」、同じ親作業を有する子作業同士を「**兄弟作業**」と呼ぶ。さらに、以下の記号を定義する。

$\mathcal{M}^u$	第1工程の機械の集合
$\mathcal{M}^d$	第2工程の機械の集合
i	親作業i
(i,j)	親作業iの子作業j
t_i	親作業iの加工日数
b_i	親作業iの加工開始から子作業(i,j)が開始できるまでの日数のjに関する最小値
$t_{(i,j)}$	親作業iの子作業(i,j)の加工日数
s_i	親作業iの加工開始日
$s_{(i,j)}$	親作業iの子作業(i,j)の加工開始日

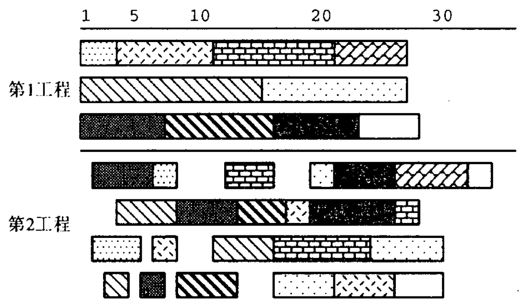

図1 スケジュールの一例

3 本研究が着目するスケジュールのクラス

本研究では、ジョブショップスケジューリングにおけるアクティブスケジュールの生成法 [1] を基礎にスケジュールを生成するが、いくつかの修正が必要となる。

3.1 開始時刻の決定とスケジュールの列挙

本研究のモデルは、兄弟作業が同じ中間品を共有するため、ある子作業を開始すると、その分中間品を消費してしまい、それ以外の兄弟作業の開始日に影響を与える。すなわち、全体のスケジュールから見た場合、子作業は必ずしも最早開始日で開始することが良いか分からず、兄弟作業の開始をそれ以上早められないという日まで当該子作業の開始を遅らせることを考える必要がある。

しかし、各機械で各作業の着手順序のみならず、兄弟ジョブの関係を考えながら各作業の開始日を決定するのは、列挙するスケジュールの数が膨大になり、現実的ではない。

3.2 スケジュールの生成方法

本研究では、1日目から始めてガントチャートを順次左から埋めるように作業を割り付けてゆく。その際「ある子作業の最早開始日が与えられたら、その日に開始する」というスケジュールにのみ着目する。子作業の最早開始日は、当該作業以外の未割り付けの兄弟作業が存在しないという前提で計算する。これは、親作

業や他の兄弟作業の開始日が分かれば中間品の量の時系列的変化が分かるので、当該子作業の中間品の払い出し量と加工日数から一意に計算できる。

3.3　割り付け対象となる作業と機械の選択

各親作業に対して、スケジュールに依存することなく、その子作業の各々の加工日数と払い出し量から、親作業を開始してからその子作業それぞれが割り付け可能になるまでの日数を得て、その最小日数 (b_i) を計算できる。親作業が開始してからこの日数だけ経過したら、少なくとも後続する子作業のうち一つが割り付け可能となることから、第1工程ではこの時刻が最早になるような作業や機械が、第2工程では最早終了日の最も小さい作業や機械が、分枝限定法の分枝の対象となる機械や作業を決定する。

すなわち、全作業の集合 T、ある部分スケジュール σ における未割り付け作業の集合を $S(\sigma)$、未割り付けの親作業の集合を $S^u(\sigma) \subset S(\sigma)$、未割り付けの子作業の集合 $S^d(\sigma) \subseteq S(\sigma)$、すでに割り付け済の作業の集合を $\bar{S}(\sigma) = T \backslash S(\sigma)$ とすると、

$$
\begin{aligned}
f_u^* &= \min\{s_i + b_i \mid i \in S^u(\sigma)\} \\
f_d^* &= \min\left\{s_{(i,j)} + t_{(i,j)} \mid i \in \bar{S}(\sigma), (i,j) \in S^d(\sigma)\right\}
\end{aligned}
$$

$$f^* = \min\{f_u^*, f_d^*\}$$

$$M^* = f^* \text{を実現する機械}$$

によって、機械 M^* で f^* 以前に割り付け可能な作業で分枝する。

4　下界の導出と分枝限定法

以下に説明する二つの下界 (LB_1, LB_2) と第1工程の各機械の加工時間の和から得られる第1工程の全作業の加工にかかる日数 LB_3 の最大値、$LB = \max\{LB_1, LB_2, LB_3\}$ を総所要時間の下界値として、分枝限定法によってスケジュールを求める。

4.1　下界 1–ジョブに基づく下界

第1工程の各機械の未割り付けの作業が、その機械で最後に加工されるときに、他に未割り付けの兄弟作業が無いという前提下での各子作業の終了日を計算することができる。第1工程の機械 $m \in \mathcal{M}^u$ で加工される未割り付けの親作業集合を $R_m^u \subset S^u(\sigma)$ とするとき、親 $i \in R_m^u$ を当該機械の順序の最後に加工し、その子作業 $(i,j) \in S^d(\sigma)$ の、未割り付けの兄弟があってもそれを無視して求めた最早終了日を $C_{(i,j)}$ とすると、

$$LB_1 = \max_{m \in \mathcal{M}^u} \min_{i \in R_m^u} \max_{(i,j) \in S^d(\sigma)} C_{(i,j)} \tag{1}$$

は下界となる。

4.2　下界 2–第2工程の機械に基づく下界

部分スケジュール σ において、第2工程の各機械の未割り付けの作業に関して、未割り付けの兄弟作業があっても無視した上で、親作業が

- 割り付け済ならば、直接最早加工開始日を計算する。
- 未割り付けならば、すでに割り付けられた部分スケジュールの後にすぐに続くように該当する親を割り付けた場合の子作業の最早開始日を計算する。

そして、未割り付けの第2工程の全作業の着手可能日が得られたら、第2工程の各機械毎に着手可能時刻付き総所要時間最小化1機械問題を解き、機械 m の最適値を C^m とすると、

$$LB_2 = \max_{m \in \mathcal{M}^d} C^m \tag{2}$$

は下界となる。

5　数値例

数値例は、FreeBSD-2.1.0R 上の gcc を用いて実装し、DELL GXMT5133(Pentium133MHz) を用いた結果である。

表1は、深さ優先探索を行なった結果だが、計算時間がかかる問題に関しては、下界の精度が悪いことが理由として考えられる。したがって、下界値の改善が高速化の鍵となる。

参考文献

[1] B. Giffler and G.L. Thompson. Algorithms for solving production scheduling problems. *Opns.Res.*, Vol. 8, pp. 487–503, 1960.

表1　数値例

ジョブ数と作業数	機械数	下界値	最適値	実行時間	評価した頂点数
(10,34)	(3,4)	30	30	0.48	236
(10,34)	(3,4)	29	30	69.45	38,689
(10,34)	(3,4)	49	50*	117.10	—
(16,57)	(5,5)	34	34	2.40	775
(20,73)	(6,6)	26	27*	356.3	—
(34,125)	(10,10)	40	40	7.01	875
(68,252)	(20,20)	51	51	80.06	4,237
(68,252)	(20,20)	42	44*	3,173	—

実行時間は CPU タイムで単位は秒。下界値は原問題に対する値。
* は、頂点を 100,000 個評価して計算を打ち切った時の暫定値。